Universidade de Aveiro 2015/2016
Departamento de Electrónica, Telecomunicações e Informática,

Jorge Miguel Sarrico Madail

Sistema de Controlo de Rebanhos
Herd Control System

Universidade de Aveiro
2015/2016

Departamento de
Electrónica, Telecomunicações e Informática,

Jorge Miguel
Sarrico Madail

Sistema de Controlo de Rebanhos
Herd Control System

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica de Paulo Bacelar Reis Pedreiras, Professor Auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro e Pedro Alexandre de Sousa Gonçalves, Professor Adjunto da Escola Superior de Tecnologia e Gestão de Águeda da Universidade de Aveiro.

o júri / the jury

presidente / president	**Prof. Doutor Alexandre Manuel Moutela Nunes da Mota** Professor Associado da Universidade de Aveiro
vogais / examiners committee	**Doutor Paulo Jorge de Campos Bartolomeu** Globaltronic - Electrónica e Telecomunicações, Sa.
	Professor Doutor Pedro Alexandre de Sousa Gonçalves Professor Adjunto da Escola Superior de Tecnologia e Gestão de Águeda

agradecimentos / acknowledgements

Um agradecimento especial aos meus pais e irmã por todo o apoio ao longo do meu percurso académico

A todos os meus colegas e amigos, que conheci ao longo destes 5 anos na Universidade de Aveiro, um grande obrigado.

Por último, um obrigado ao meu orientador, Professor Doutor Paulo Bacelar Reis Pedreiras, e co-orientador, Professor Doutor Pedro Alexandre de Sousa Gonçalves, por toda a ajuda e disponibilidade prestada ao longo deste trabalho.

Resumo

O modo como as vinhas se encontram organizadas faz com que seja necessário remover as plantas mais rasteiras que competem com as videiras por exposição solar e nutrientes. Até agora, o uso de máquinas e herbicidas tem sido a solução adoptada, mas estas técnicas são agressivas para o ambiente, sendo necessário encontrar outras alternativas.

Ao longo dos anos várias técnicas foram desenvolvidas para controlar rebanhos, uma vez que estes animais podem substituir o homem em diversas tarefas, como por exemplo, limpeza das florestas. Este tipo de animais consegue chegar a sitios onde algumas máquinas não conseguem e a vegetação presente faz parte da sua dieta.

No âmbito desta dissertação tomou-se como base o conceito de cerca virtual, desenvolvendo-se um conjunto de novas funcionalidades que têm por objetivo adaptar este tipo de tecnologia às caractersticas de culturas específicas, como por exemplo a vitivinicultura. O trabalho realizado versou quer a vertente tecnológica, nomeadamente tecnologias de sensorização e localização, que a vertente económica, tendo-se neste caso por objetivo obter um sistema de baixo custo, que possa ser massificado.

Abstract

The way vineyards are organized brings up the need of removing the creepers that compete with vines for solar exposition and nutrients. Until now, the use of machines and herbicides has been the adopted solution but these techniques are aggressive for the environment and for this reason, new alternatives have to be found

During the years several techniques have been developed to control herds, once these animals can replace the man in different tasks, for example, forest cleaning. This type of animals can reach places where machines can not and the present vegetation is part of their diet

This dissertation builds up on the concept of virtual fence, extending it with a set of new functionalities that aim to adapt this type of technology to the characteristics of specific crops, such as viticulture. The work carried out has focused both on the technological aspects, such as sensing and localization technologies, as well as on the economic aspects. In this latter case, the aim is to obtain a low cost system that can be massified.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Every year, especially in summer, many countries suffer from the effects of forest fires and one reason why these gain high proportions is that almost all forests are not clean. Using manpower for cleaning forests is too expensive and some animals can do that job. Goats are the most adequate animal to perform this activity. They are robust animals that can adapt to different types of terrain, reaching areas unaccessible to machines, and their diet includes vegetation that exists in abundance in forests. Some years ago, a project called "Self-Prevention" [6] was presented with the objective of placing these animals in abandoned grounds in the Douro region for them to clean those lands, and so prevent forest fires, and also study the effects of the presence of goats for a period of time in those terrains. Unfortunately this project was not yet implement due to lack of financial support.

This dissertation will focus in vineyards maintenance. Over the last years, vineyard owners have been using plows and herbicides, which are very aggressive for the surrounding environment, what leads to a degradation of the product quality. The use of sheeps to perform this activity has several advantages such as environmental impact reduction and terrain fertilization but these type of animals can not be used for this task during the entire year because they will tend to feed from the lower vegetation and fruits, which damages the production. So, it is necessary to develop a system to control the animals during this part of the year.

There are several solutions available on the market to control this type of animals, but none is suitable for these activities. Those systems are called virtual fences and are used only to keep animals in specific zones. We intend to develop a system capable of maintain animals in a predefined area without the use of physical elements like fences or electric wires and also with the ability of control the posture of animal in order to prevent them from feeding from the cultures.

It is known that the production of wine is one of the most important business areas in our country. In 2014, the area of planted vineyards was about 218.667 hectares [7], what makes us expect the success of this project.

1.2 Objectives

The control of all animal's movements is the main objective of this dissertation. We intend to know, in real time, the localization of each animal in a predefined area and gather information about animal's body behavior, for example, if it is lying down or not.

To determine the posture of the animal, a combination of sensors is used to measure the inclination of animal's neck as well as its distance to the ground. The system also comprises an electrostatic actuator that penalizes the animal when it adopts unwanted behaviors. This way, if the animal tries to feed itself from the cultures or if it moves out of the designated area, it receives a penalization that discourages its behavior

The localization system requires the implementation of a communication protocol between a fixed pole and the animals. Since we intend to make this system affordable both in economical and power consumption terms, we must design it using an energy saver schema.

The final product will be a collar to place on the animal's neck that contains these three systems. Each collar will communicate with a fixed pole and the herd manager will be able to gather information from it.

In resume, the objective of this dissertation is to improve the actual virtual fencing concept by adding new features and improve the ones that have been already implemented.

1.3 Structure

This dissertation is composed by the following chapters:

- Chapter 2, "State of the Art", where we present the important concepts that we will use in this project. It starts with virtual fencing and then localization methods and rf modules, distance measurement techniques using ultrasonic sensors and finally some information about accelerometers.

- Chapter 3, "System Architecture", where it is explained how the system will be implemented, which approach we took and how it is divided.

- Chapter 4, "System Implementation", divided in three sections. In the first one, distance to ground measurement, we explain in detail the developed hardware and firmware to interface the ultrasonic transducer. The second is tilt angle, where we present the accelerometer used and how we obtain the angle the head is tilted. Finally, the last section addresses the communication protocol and contains all information about the firmware to interface the RF module with the microcontroller and also the communication schema that we developed.

- Chapter 5, "Results", contains all the tests carried out for all parts of the work and respective analysis.

- Chapter 6, "Conclusions and Future Work", where we do a critical analysis to all project and point out some features or improvements that can be done in the future.

Chapter 2

State of the Art

As explained before, the objective of this dissertation is to develop an embedded system to control animal's position and behavior. We plan to improve the solutions that already exist on the market by adding new features and also make them more affordable, both in production cost and power consumption.

First we will look to the concept of virtual fencing, which is the basic idea for this project and analyzing in detail the several methods of localization and comparing them in terms of efficiency and adequacy to the environment were we want to use them. Next we describe ultrasonic sensors and accelerometers, what they are used for and we do some market research to see which ones are more suitable for this work.

2.1 Virtual Fencing

Conventional fences are static tools that are very effective in controlling animal ingress or egress but fail to offer managers the flexibility they need to optimize the physiological requirements of the vegetation with the nutritional needs of foraging animals [8].

To solve this problem, over the years some solutions have been developed and the concept of **virtual fencing** [9] came up. It consists on establishing a virtual zone and with suitable localization methods and sensors to record information, actuate to prevent animals from entering or exiting that area.

There are several methods to guard animals using virtual fences. One of them consists in placing an isolated electric wire underground in which an electromagnetic signal is applied (Figure 2.1). Each animal carries a collar with a receiver that detects the presence of the wire's electromagnetic field and actuates to avoid animal passing through. This approach only solves the problem of physical barriers because it does not provide any other information about animals and the cost of implementation and maintenance is very high.

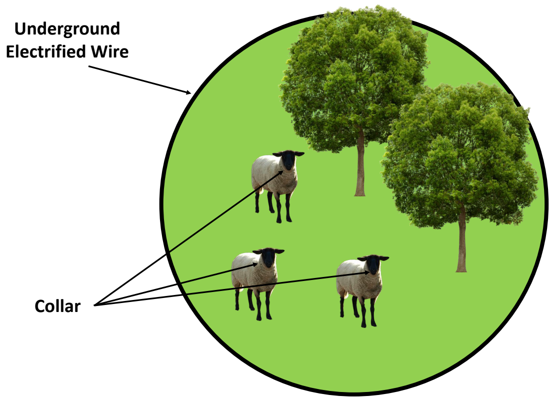


Figure 2.1: Conventional Virtual Fence

Other solution to guard animals is Global Position System (GPS). Each animal is equipped with a collar that includes a GPS unity and actuators. The area is defined by GPS coordinates and the position is verified periodically (Figure 2.2). When the animal is near the limits an actuator induces a stimulus. The presence of a GPS unity brings up some problems such as power consumption and autonomy. Moreover, the lack of precision of GPS causes difficulties on the animal's learning process, decreasing the system effectiveness[9].

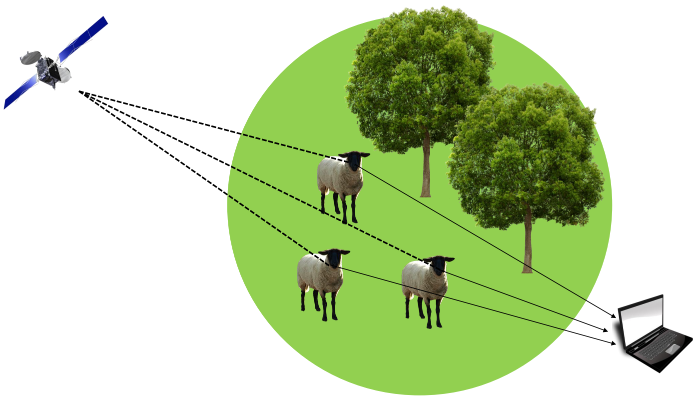


Figure 2.2: GPS based Virtual Fence

The methods available on the market solve the problem of animal confinement but only with GPS localization they become portable. But for this type of projects, the power consumption is a very important issue and because of that, GPS is not a good solution. The ideal virtual fence is the one that allows the farm manager to know the relative position of each animal, get information in real time about the state of each one and with some additional sensors, control the posture of animal in order to avoid damage to important resources. At this moment, no such solution is available.

2.2 Localization using Radio Systems

Object position can be obtained through the use of radio waves. This type of localization consists on transmitting signals between RF modules. Is applied in real-time systems and processes wave characteristics. It is widely used and there are several methods available depending on the type of environment, accuracy and number objects to locate.

Usually, this technique is applied in Wireless Sensors Networks [10] since it does not require additional hardware and can be applied in low power devices, common in this type of networks.

2.2.1 Methods of Localization

Through the use of radio waves it is possible to gather information related to those waves such as power and time. This requires at minimum two nodes (each node corresponds to a different device) in order to have signal transmission and with that is possible to get the power of the received signal and calculate the time waves take to travel from one node to the other.

With this information we can choose between several techniques:

- **Received Signal Strength Indicator (RSSI)** - indication of the power level being received by the antenna: the higher is the RSSI level, the stronger is the radio signal and closer to the destination [11]. To use this method, the RF module used needs to be able to measure the received signal power. One of the reasons that make RSSI a very good technique is the fact that the power level is sent during packet transmission, what makes this very simple and low cost.

  But this method has some drawbacks. The efficiency is affected by path loss, fading and shadowing. **Path loss** is the reduction in power density of an electromagnetic wave as it propagates through space [11]. The amount of power decrease depends on type of environment. In free space, this reduction is smaller than if we have buildings or in indoor spaces. **Fading** is deviation of attenuation that a signal experiences [11]. This occurs when signals from different paths collide. Collision could be constructive or destructive and so amplify or attenuate the signal power at the receiver. **Shadowing** is the loss of signal due to obstacles (walls, buildings, trees, cars, people, etc.) between a transmitter and a receiver [11].

  To compute the distance using RSSI several algorithms are available but for this type of project two of them are the most suitable. The first one consists on obtaining empirical values of RSSI for each value of distance and create a database. The other one uses the following mathematical equation [12]:

  $$Pr = Pt * (\frac{1}{d})^n \tag{2.1}$$

  With some simplifications, equation (2.1) can be written as [12]:

  $$10logPr = 10logPt - 10nlogd \tag{2.2}$$

  Where Pr is the value of RSSI, Pt the transmitted power, n the signal propagation constant and d the distance between transmitter and receiver.

- **Time of Arrival (ToA)** - consists on measuring the absolute time that a signal takes to get from the transmitter to the receiver. RF signals travel at the speed of light, so one meter accuracy requires approximately a 3 ns time resolution[13]. For some devices this value is not reachable because they are limited to their reference clock. The efficiency of this approach is affected by noise, time synchronization issues and multi-path channel. These problems could be attenuated by the use of additional hardware, increasing the complexity, cost and power consumption of devices.

2.2.2 RF Modules

As seen before, to get the relative position of some devices, signals have to be transmitted between two nodes (minimum) and to perform that, RF Modules are required.

RF Modules are small electronic devices used to communicate with others using radio frequency. These type of modules can be applied in many areas, for example, wireless communications, domotics, security systems and robotics Each device incorporates a transmitter or receiver circuit, an antenna and interface to host processor. There are three types of RF modules:

- **Transmitters** - modulate and transmit a signal to carry data. Usually, a microcontroller can be introduced to define the information to send.
- **Receivers** - receive and demodulate the signal. They can be divided in two categories: **super-heterodyne** and **super-regenerative**. The differences between these types of receivers is that super-heterodyne are more expensive and operate in a wider frequency range than super-regenerative, because the first type incorporates a crystal oscillator that defines the operating frequency and the second contains a feedback loop that acts as a quench oscillator[14].
- **Transceivers** - include both transmitter and receiver on the same module.

Figure 2.3: RFM42B-868D RF transmitter module [1]

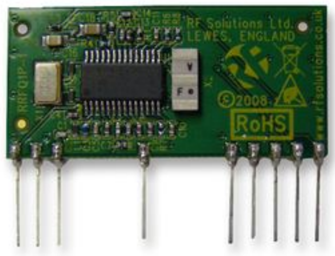


Figure 2.4: FM-RRFQ1-433 RF receiver module [2]

Some modules include their own microcontroller that handles packet transmission but others require a host microcontroller to process the information so an interface is required. UART, SPI and USB are the most common interface types. More recently, Bluetooth is also an option, for example, nrf51 modules from Nordic.

The applications where these modules are used require them to be low-power, cheap and small size. With the growth of Wireless Sensor Networks [10], the use of these type of devices also increased and each network incorporates up to hundreds of RF modules. In most cases the use of transceivers is suitable because each device has to receive and transmit data and for that reason, having both on same chip makes it smaller and power consumption is lower. For transceivers, the most used operating frequency bands are 2.4GHz and 900MHz.

RF modules are widely used in many applications because unlike optical devices such as Infra Red (IR), they do not require line-of-sight to communicate.

2.3 Distance measurement techniques

In this project we intend to develop a system capable of measure the distance between animal's head to ground. In this subsection, we will point out the available techniques to perform this and which type of sensors can be used.

Distance is obtained by wave propagation time, so it is possible to use ultrasonic or electromagnetic waves. The use of the first type has some advantages when comparing to the electromagnetic waves. For example, measuring travel time and differences of phase between signals is relatively easy and not expensive, because the speed of ultrasound propagation is several orders of magnitude lower than the electromagnetic waves [15].

The information above leads us to conclude that the best solution is to use ultrasonic waves. Since we will transmit and receive them, we have to pay attention to the surrounding environment where we intend to use it. Speed of ultrasound propagation and its range depends on the air temperature, relative humidity and air pressure. Air temperature changes speed of ultrasound propagation by the relation of 0.17 %/°C at the temperature of 25 °C. This value cannot be neglected. Low temperatures increase the detection range of sensors and that raise is independent of the relative humidity [15].

As said before, distance is determined by processing the propagated ultrasonic wave between transmitter and receiver. It is possible to compute the travel time or the phase difference of the signals and so, we can have two techniques: **Continuous wave** or **Pulsed wave**.

2.3.1 Continuous wave

This method is based on the comparison of the emitted signal phase with the phase of the received signal. An oscillator generates a signal with a certain frequency and that signal is applied to the emitter and read by the receiver. As they are separated by a distance, the emitted and received signals are both applied to a phase detector. A comparator turns the output signal of the phase detector, corresponding to the phase difference between both signals, into a signal with only two levels, generating a square wave. The phase detector output changes when one of the transducers move. This output value is maximum when both signals have the same phase and minimum when they are in phase opposition. The movement of one transducer generates a bunch of impulses in the comparator output, starting a counter. If that movement occurs in the same direction, the displacement is equal to wave length, which corresponds to the ratio between the sound speed propagation of sound, c, and the frequency of the signal emitted [15]. In Figure 2.5 is represented a diagram that describes this technique.

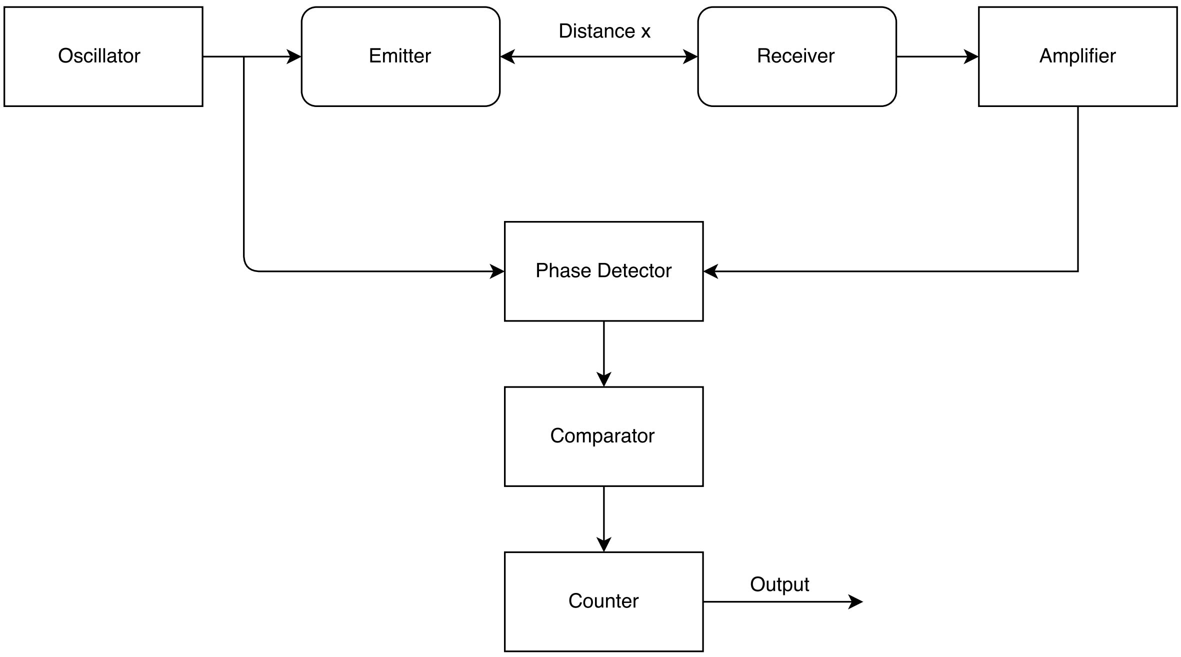


Figure 2.5: Continuous wave method to measure distances

2.3.2 Pulsed wave

On the pulsed wave method, a burst of impulses is emitted and the distance is obtained by measuring the travel time of the reflected pulses. A generator produces a signal and in the active period it emits a burst of impulses and also initiates a counter. When that burst of impulses reaches the receiver, it is generated an impulse that stops the counter. The ratio between the output of the counter and the oscillator frequency corresponds to the travel time. With that value, the distance is calculated by multiplying it with the sound propagation speed in the air. This is used when the emitter and receiver are located face to face and separated by a given distance.

Figure 2.6 shows the time diagram of this technique. In some cases, the reflection is made by an object and the emitter and the receiver are side by side (Figure 2.7). The process of measuring the distance to that object is the same as before, but the travel time is half. This requires a good ultrasound reflector object and its position must guarantee that the reflection reaches the receiver.

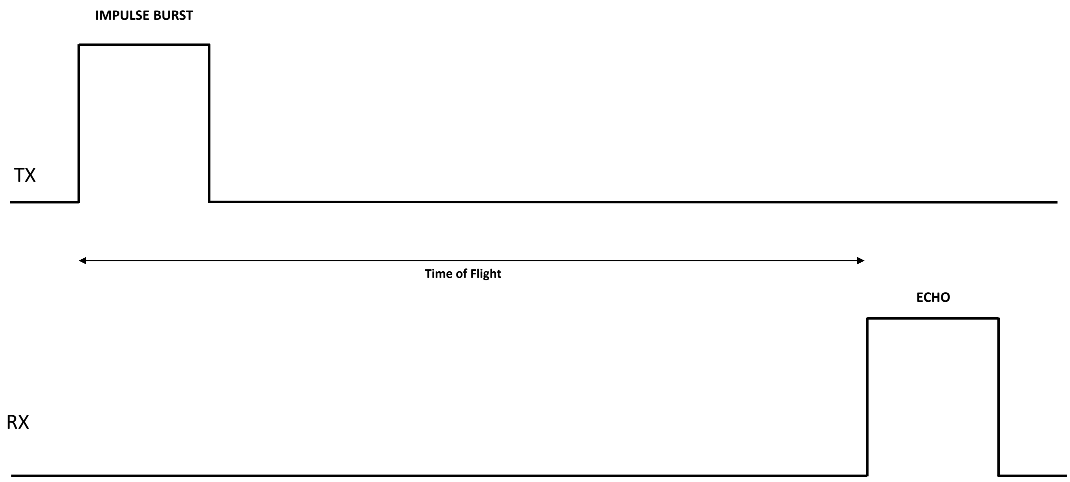


Figure 2.6: Time diagram of pulsed wave method

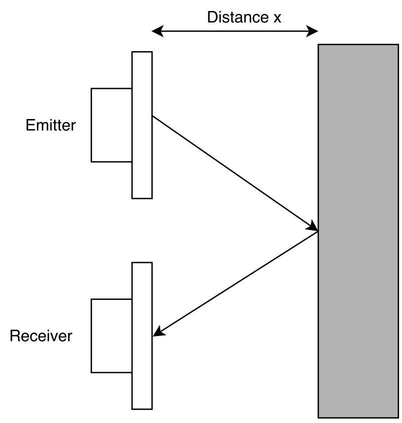


Figure 2.7: Ultrasonic sensor with emitter and receiver side by side

For this project, the pulsed wave method is the most suitable. We only have to measure the time of flight and then compute the distance from that. The continuous wave method cannot be used in our case, as it measures relative movement, not absolute distances, as it is required.

2.3.3 Utrasonic sensors

Ultrasonic sensors are used for a wide variety of applications. This is possible because these sensors can detect materials with different natures, for example, solids or liquids. The properties of objects such as color, texture and transparency do not affect the quality of detection and sensors can work in critical conditions.

For the project we intend to develop, we need a sensor capable of working outdoor, where environmental conditions may vary as well with different types of terrain. When choosing the most appropriate sensor, we must ensure that it is waterproof. After some research, we are able to present a list of some ultrasonic sensors available on the market and in Table 2.1, were we point out their most important characteristics:

Table 2.1: Table containing the characteristics of some ultrasonic transceivers available on the market.

Reference	**Manufacturer**	**Ressonant Frequency (kHz)**	**Operating Temperature (°C)**	**Waterproof**	**Price (€)**
400EP250	Prowave	40	-30 to 80	yes	14.06
400EP18A	Prowave	40	-30 to 80	yes	15.80
400PT160	Prowave	40	-30 to 80	no	4.52
400PT120	Prowave	40	-30 to 80	no	9.37
400EP14D	Prowave	40	-30 to 70	yes	19.70
MCUSD16A40S12RO	Multicomp	40	-30 to 85	yes	3
MCUSD14A40S09RS-30C	Multicomp	40	-40 to 85	yes	5.88

The ultrasonic transducers presented in Table 2.1 were found in Farnell website. As said before, we need to choose one that is waterproof, because it is intended to be used outdoor. Looking at the table, only three are waterproof and these are the more expensive ones. We want this system to be affordable, so we can not spend this amount of money for one sensor.

In this project, we intend to test if the use of an ultrasonic sensor is suitable for this application and so, we decided to use a car parking sensor. Since we do not have any information about the used sensor, we made some tests to get some characteristics of it. In following chapters, we will describe in detail how we tested the sensor and then we present the characteristics of it.

2.4 Inclination Sensors

The control of the animal's posture is one of the objectives of this dissertation. We intend to develop a system capable of detecting the behavior of the animal in order to avoid unwanted movements from it. This topic has been already studied in several projects, for example, [16] and [17], where accelerometers are used to detect the movement of animal's body. In this project, we intend to develop a similar system but adding another type of sensor, an ultrasonic transducer, to fuse information from both and improve the accuracy of movement detection. This will allow us to define several positions and then decide if the animal is behaving correctly or not.

In the subsection below, we will describe in detail some important characteristics of accelerometers and then do a market search to decide which one suits better for this dissertation.

2.4.1 Accelerometers

Accelerometers are devices used to measure acceleration forces. These forces can be static, for example, the gravity force or can be dynamic, like movement or vibration [16]. This ability allows detecting the angle that a device is tilted and also the type of movement of it, which makes accelerometers very useful in a large amount of applications today [18].

Smartphones and tablets include accelerometers to decide how the screen is displayed, aircrafts to maintain the correct direction, cars to measure acceleration and inclination. The capacity of measuring vibrations make this kind of devices very useful to detect seismic activity for example.

There are several ways to implement an accelerometer: using piezoelectric or piezoresistive effect, with a capacitance and with heat transfer.

- **Piezoelectric effect** - consists on a microscopic crystal that gets excited by acceleration forces, which generates a voltage [19].

- **Piezoresistive effect** - similar to the piezoelectric but in this type, it generates a change in the resistance. Most of piezoresistive sensors are used to measure pressure [19].

- **Capacitance** - two micro structures create a capacitance between them and with the movements of the accelerometer, they become closer or further apart, changing the capacitance value. Then, with some electronic, an equivalent voltage value is obtained. Capacitive accelerometers are also less prone to noise and variation with temperature, typically dissipate less power, and can have larger bandwidths due to internal feedback circuitry. [20].

- **Heat Transfer** - composed of a single heat source centered in a substrate and suspended across a cavity. They include equally spaced thermo resistors on the four side of the heat source [20]. Variations of the acceleration leads to changes in the heat value.

To process the data from accelerometers, a microcontroller is required and for that reason, interface with the outside world is important.

Analog or **Digital** is one of the most important characteristics and it depends on microcontroller. The first one gives us a continuous voltage proportional to acceleration and the second usually returns a Pulse Width Modulation (PWM), where the acceleration value is proportional to the time voltage is high. Digital output is more complex to work because it involves signal processing to extract the duty cycle value. On the other hand, analog output only needs a microcontroller with an analog to digital converter (ADC). There are also accelerometers that allow digital communication interfaces (eg. SPI, I2C), such as the ADXL345.

The **number of axis** depends on the task we want to perform. Is possible to have 2 or 3 axis. If 3D position is desired, one accelerometer with 3 axis is more suitable even though is possible to reach that with two devices of 2 axis. The **output range** is also important and for simple projects like measuring the tilt of something, a range of ± 1g is enough but for others, ± 5g or more could be necessary.

One issue related to the use of accelerometers is accuracy and to improve it, some methods must be applied. In datasheets, is possible to find certain variables that indicate some important characteristics for specific device such as **offset** and **sensitivity**. The first corresponds to output voltage level when no acceleration is applied and the second is the amount of voltage change per unit of acceleration. These values are dependent of device.

Is necessary to calibrate the accelerometer and to do that, offset is used. For each reading, this value is subtracted. But each reading is affected by noise and to solve that, some filters must be applied. One good solution is to take several measurements and use an average value.

For this dissertation, we need to choose one accelerometer and for this reason, we did a market search and we present, in Table 2.2, a few accelerometers and their more important characteristics.

Table 2.2: Technical specifications of some accelerometers available on the market

Reference	**Number of Axis**	**Output Interface**	**Range (g)**	**Sensitivity**	**Bandwidth**	**Price (€)**
ADXL345	3	Digital - I2C or SPI	2,4,8,16	2g - 256 LSB/g 4g - 128 LSB/g 8g - 64 LSB/g 16g - 32 LSB/g	user selectable from 3.125 Hz to 1600 Hz	11.07
ADXL335	3	Analog	3	300 mV/g	user selectable from 1 Hz to 500 Hz	5.56
DE-ACCM3D2	3	Analog	2	720 mV/g	500 Hz	No Information
FXLN83xxQ	3	Analog	2,4,8,16	2g - 256 LSB/g 4g - 128 LSB/g 8g - 64 LSB/g 16g - 32 LSB/g	High: 2.7 KHz for XY axis and 600 Hz for Z axis Low: 1.1 KHz for XY axis and 600 Hz for Z axis	2.60

Looking to the characteristics of the accelerometers presented above, we can see that most of them include analog output interface. This requires that the used microcontroller includes an analog to digital converter. For the work we intend to perform, we do not need to have an accelerometer with high bandwidth and with more than $\pm$ 3g range. These project requirements makes us conclude that, from the Table 2.2, the best solution is to use the ADXL335.

Chapter 3

System Architecture

With this project, we intend to improve the virtual fencing concept by adding new features that allow managers to have a better control over their animals. With these improvements, animals can be used for other activities that before were not possible. We intend to develop a system capable of determine each animal position, control its posture to ensure that they do not damage productions and that allows managers to receive, in real time, specific information about every animal. Since we are dealing with a high number of animals, we must ensure that this system is affordable in terms of cost and is efficient in terms of power consumption.

Figure 3.1 represents an overview of the desired system. A fixed pole broadcasts a synchronization message and each animal will receive it and transmit information back to the pole.

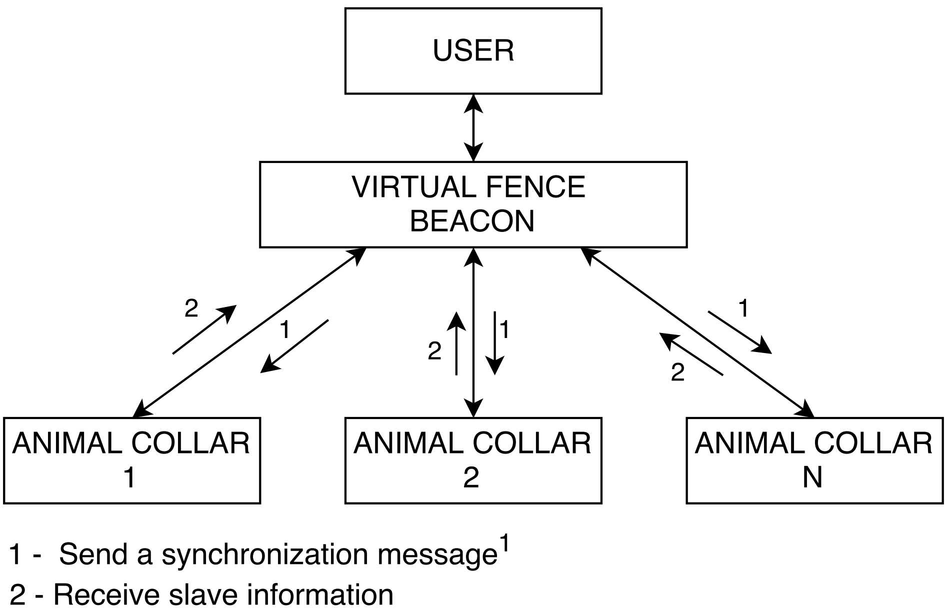


Figure 3.1: System Schematic

This work can be divided in three parts: the communication between animals collar and the base station, a set of sensors for posture control placed in the collar and then the base station and the user interface.

For the first one, we need an efficient localization system. After some research, we came to the conclusion that the best solution is to use a radio localization method. For the many available, RSSI is the most suitable for this project because it does not require additional hardware, can be used outdoors and it is affordable.

Posture control means adding some sensors to detect the movement of animal's body. Initially we only intend to determine head's position in order to prevent animals from damaging the cultures. We need to know the distance from animal's head to ground and also the inclination of it. Since we are dealing with outdoor systems, choosing sensors that can operate in these conditions is extremely important.

For this dissertation, to prove the concept, only two slave nodes are used but in future, it shall be possible to monitor a higher number of animals, which may require the use of more than one pole in different positions. This allows us to increase the area of control. These devices have to be powered and portable, so we will include a battery in each collar, and its charge level is one part of the information that each collar sends to master. For this one, and once it will be in a fixed location, we have several ways to power it, for example, with solar panel.

Each animal incorporates a collar, which is shown in Figure 3.2. Each one includes a microcontroller, an ultrasonic sensor, an accelerometer, an electric stimulator, a warning buzzer and a RF module.

The collar has to be able to compute the data from both ultrasonic sensor and accelerometer and decide if has to actuate or not. It includes a microcontroller responsible for the referred task and also a RF transceiver to communicate with the pole, in order to send and to receive current status and also to measure the distance between both to obtain the position of the collar.

An electric stimulator and warning buzzer will be part of the system but in this dissertation they will not be implemented. The idea of both is to apply electrostatic impulse and an audible sound to make animals learn what they can do and what they can not.

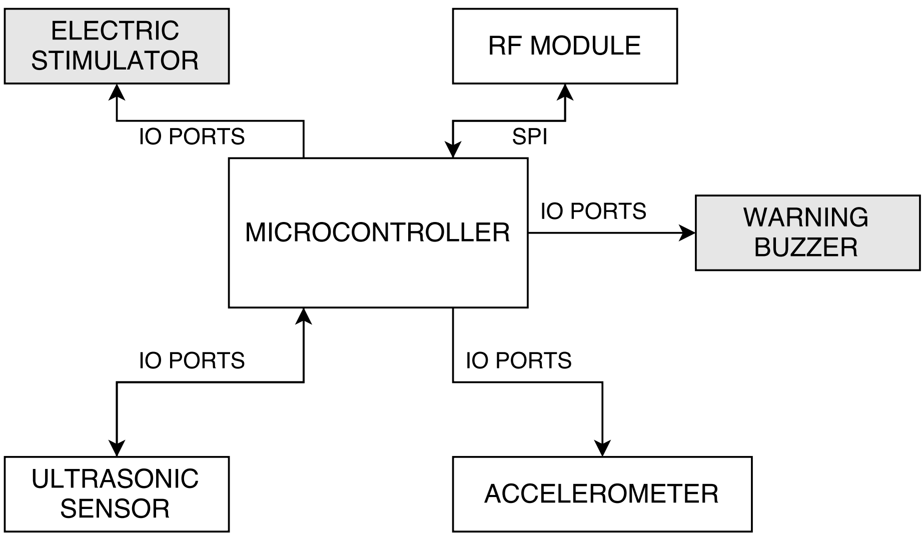


Figure 3.2: Collar Block Diagram

The ultrasonic sensor is used to measure distance of the head to the ground and an accelerometer to detect the angle that the head of the animal has with respect with the ground. The use of this last device is to increase the accuracy of the ultrasonic sensor, because there will be situations where distance given by this sensor does not allow us to detect if the animal is in a risk position or not. With the fusion of both sensors, we can define various situations and then actuate with more reliability.

The localization part is done with an RF module. The device used includes a SPI interface and it has to allow access to the RSSI value to let us compute the distance between the collar and the pole.

We have to implement a communication protocol between the master (pole) and the slaves (collar). We must ensure that we have no collisions to minimize power consumption. For these reasons, we decided to use a Time Division Multiple Access (TDMA) schema. The objective is the master broadcast a synchronization message to the slaves and then, each slave is attributed a specific time slot and transmits a pre defined message back to master. Since we can get data from both ultrasonic and accelerometer sensors, the message has to include them. The master has to know also which slave is transmitting and for that reason, we will define message with slave ID, the distance and the angle.

This protocol, allows us to have a high number of slaves communicating with a single master, what reduces the cost of the intended system. To complete this implementation, we have to define the width of each slot. Knowing the size of the messages, we can determine the transmission time and this value is the base to define the slot time.

In Figure 3.3 is represented the time diagram of the intended communication protocol.

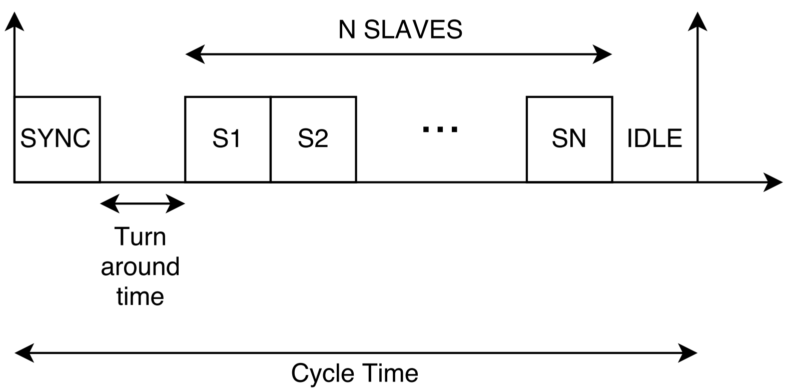


Figure 3.3: TDMA time diagram

After the transmission of the synchronization message, we have to set a turn around time which corresponds to the time that the modules need to change from transmission mode to receiving mode. After that each slave transmits its data within the respective time slot and enters into sleep mode, until the beginning of the next cycle. In turn, the master node receives the information from all nodes, processes it and also enters into sleep mode until the beginning of the following cycle.

Chapter 4

System Implementation

In this chapter we explain in detail how each part was implemented. We start by developing the system responsible for measuring the distance to the ground. We will describe in detail the circuit to interface the microcontroller with the ultrasonic sensor, how it is divided and some specific parts. For the other tasks, we did not develop any hardware since all modules used include the necessary circuits.

We will also present, with the help of some diagrams, the firmware developed for the different parts of this project.

4.1 Distance to ground measurement

To prevent animals from reaching the highest vegetation is necessary to develop a system capable of measuring the distance from neck to ground and actuate when that value corresponds to an unwanted situation.

In the state of art, we presented a table with some ultrasonic sensors available on the market. Once this project is intended to be low cost, we decided to use a sensor similar to the ones used in car parking systems (Figure 4.1). We know they are waterproof and can operate in the same conditions that we need them.

This sensor is cheaper than the others and it has the required features.

Figure 4.1: Ultrasonic sensor used [3].

This system is intended to be used outdoor and we must ensure that it is capable of measuring in different terrains, with higher or lower vegetation, good or bad weather (Figure 4.2).

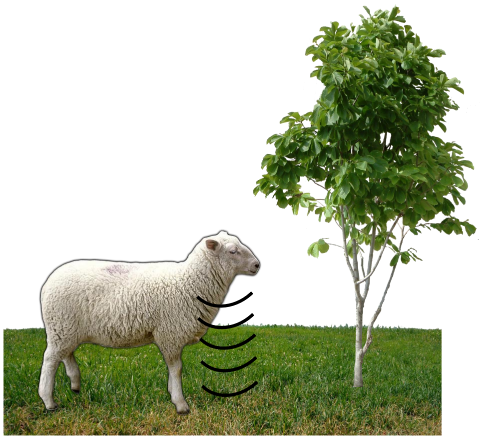


Figure 4.2: Head to ground distance measurement

As referred before, this part of the project requires hardware and software development, so below, both will be explained separately.

In Figure 4.3 the developed circuit is described using block diagrams. For our case, it consists in two parts: transmitter and receiver. If we did not have any microcontroller capable of generating the necessary burst of impulses, we needed to implement an oscillator.

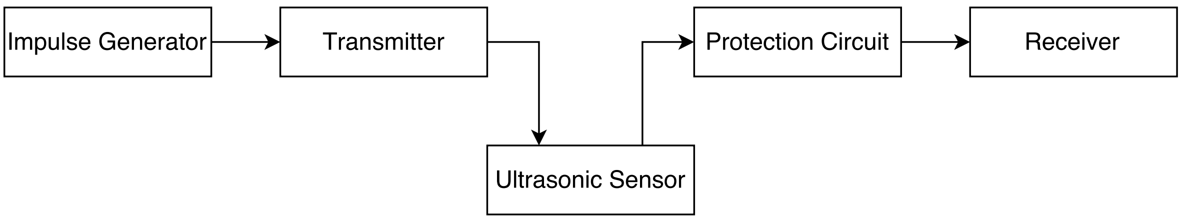


Figure 4.3: Hardware Blocks Diagram

The transmitter includes a sensor drive circuit. It is used to generate the voltage to drive the ultrasonic sensor. We know that this type of sensors requires high voltages to operate, approximately 100V, so it includes a transformer with 1/10 ratio. Since our voltage source is 12V, we thus supply our sensor with enough voltage.

Since the same transducer is used for both transmission and reception, during transmission more than 100V are applied to the reception block, which is designed to amplify very small voltages and is supplied with 5V. To prevent damages to the receiver circuitry, a pair of clamping diodes is placed between the transducer and the receiving circuitry (Figure 4.4).

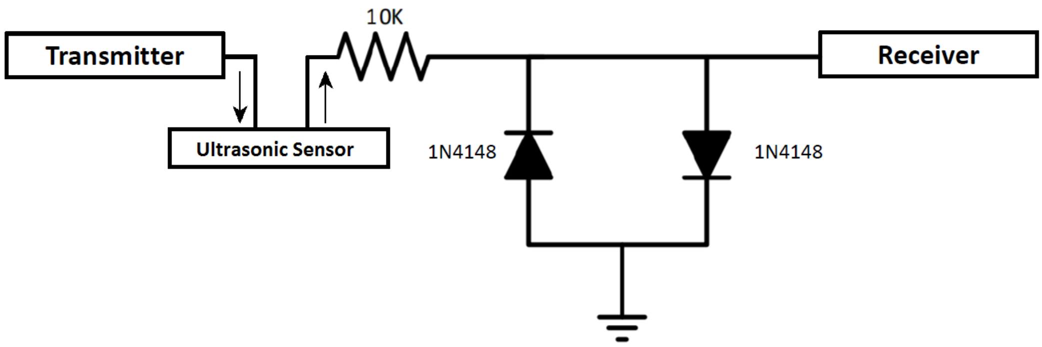


Figure 4.4: Protection Circuit

With the transmission done, we have to receive and process the echo signal that reaches the sensor back. First we have to amplify it, and to do that, we use two amplifier stages. The first stage has a 35dB gain and the second 20dB. These values were defined by empirical tests and allows us to measure in the desired range, from 40 to 100cm. After that, signal passes to a peak detector and finally to a comparator, as illustrated in Figure 4.5.

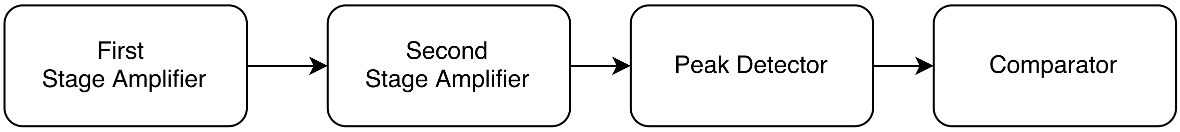


Figure 4.5: Receiver blocks diagram

The peak detector (Figure 4.6) corresponds to a half wave voltage doubler with similar rectifier behavior. This type of circuit is used to produce an output two times higher than the input, but in our case we do not want to have that increment of voltage, we only need to obtain a signal that can be used for the comparator and also to reduce noise. To do that, we have to choose the most suitable capacitors and load resistance in order to reduce the ripple, which is done by using large value capacitors.

At negative cycle of sine wave, diode D1 is forward biased and conducts charging capacitor C1, which has no path to discharge, and so it works as a storage device. During positive cycle, D1 is reversed biased and again C1 can not discharge, but D2 is charging up C2. Since C1 has a voltage equal to input value across it, C2 charges twice that value. With that, at the output of this circuit we can have double input voltage.

The value of C1 and C2 produces an output value similar to input and the signal rectification is more efficient and stable than using conventional half wave rectifiers with only one capacitor and diode. The load resistance is used to discharge the value stored at C2 and should be higher than 100kΩ.

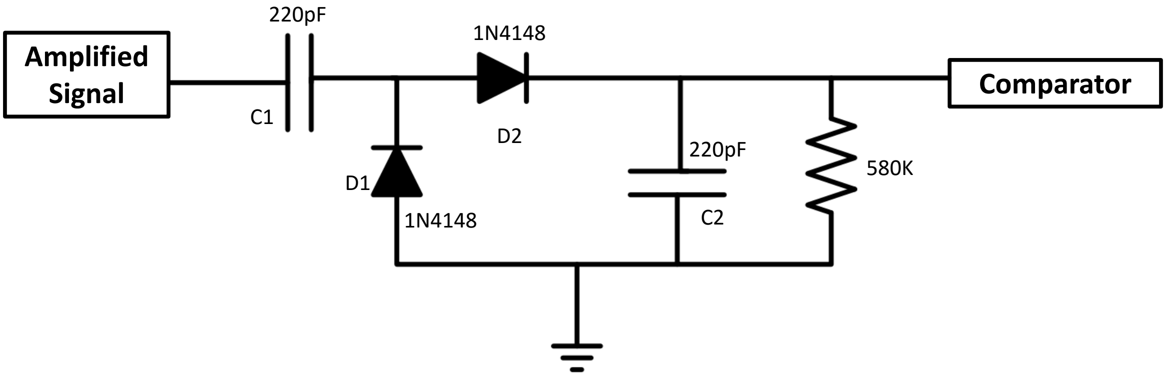


Figure 4.6: Peak Detector

The last part of the receiver is a comparator. The objective of this circuit is to detect the echo. The rectified signal is compared to a reference voltage and the output of the comparator is connected to the microcontroller. This reference was determined by empirical test and we decided that 1.2V allows us to obtain values in the desired range. The complete electrical scheme of the hardware can be found on Appendix A.

With the hardware implemented and tested, it is necessary to develop the firmware. We have to generate a burst of impulses, transmit them and then measure the time of flight when the received signal reaches the sensor again.

Figure 4.7 depicts a flowchart that describes this part of the firmware.

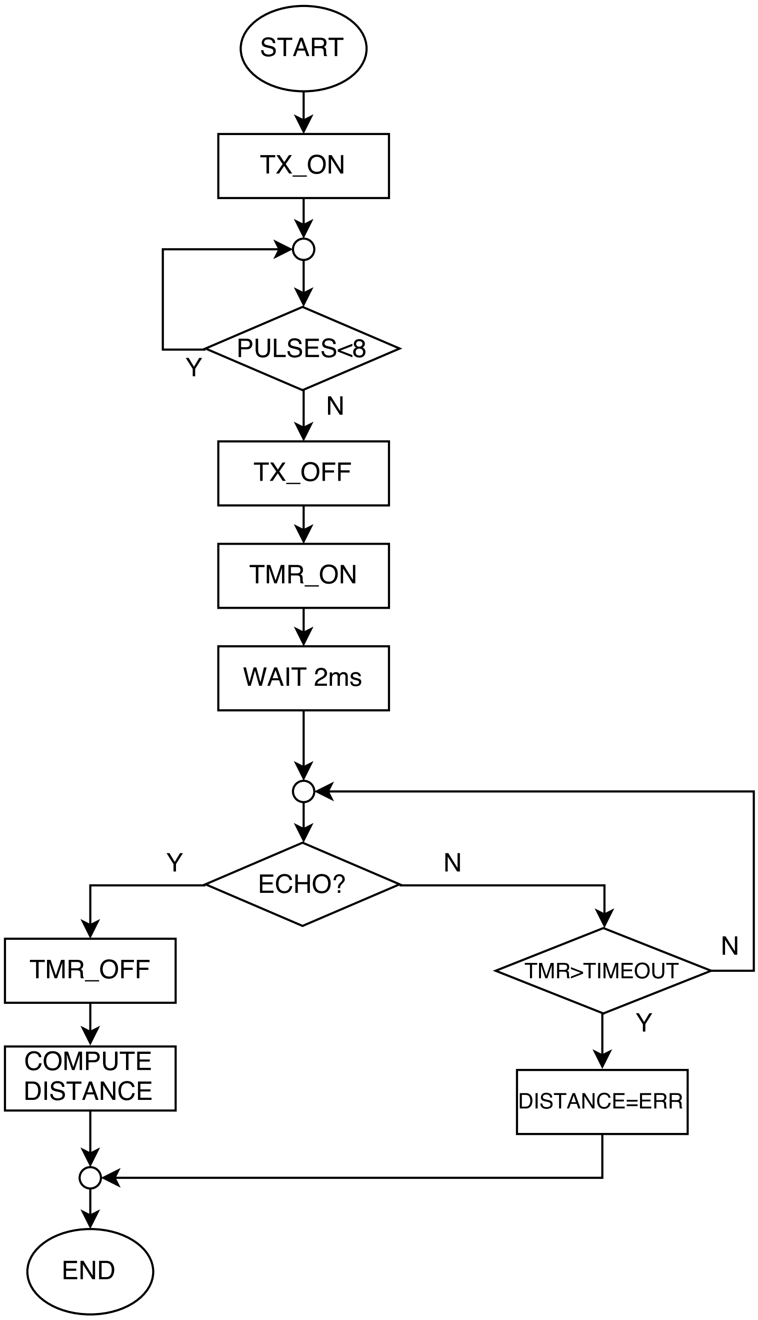


Figure 4.7: Distance Measurement Firmware Block Diagram

As explained before, to measure distances with ultrasonic sensor is necessary to generate a burst of impulses and then measure how much time passed between transmission and reception.

The resonant frequency of the used sensor is approximately 41KHz and with some tests we conclude that an impulse of 250us is enough to the intended results, what corresponds to 8 pulses. The microcontroller used, which is a PIC18F258, contains a PWM module and we used it to create a square wave with the desired frequency and duration. Before transmitting, timer starts to count and we set the state to NOT_ECHO and all interrupts are disabled.

After the transmission it is necessary to wait about 2ms to ensure the initial received signal is faded, since the transmitter is a resonant circuit. When then echo arrives, in the interrupt routine, the state changes to ECHO, the timer is disabled and the distance is obtained from the following formula:

$$Distance = Time * Speed\_of\_sound \tag{4.1}$$

It is known that distance is a relation between speed and time, so we have to know the sound propagation speed and the wave travel time. Since we are using a timer to count, is important to know in detail all timer configurations.

$$Period = \frac{1}{Fosc} = \frac{1}{5MHz} = 0.2us \tag{4.2}$$

Now we have to multiply period with the configured prescaler:

$$Period = 0.2 * 256 = 51.2us \tag{4.3}$$

With this value, is possible to convert **TimerValue** to a time unit. For simplicity, we will use microseconds.

$$Time(us) = TimerValue * 51.2 \tag{4.4}$$

As we know, the objective is to obtain the distance and now have all we need to calculate that value. We know the pulse-echo time and the propagation speed of the ultrasound. This last value is an approximation because it varies with environment conditions, such as temperature and humidity. An ultrasound travels at approximately 340 m/s, which gives 34 000 cm/s. Since we intend to measure the echo time, this value has to be divided by two, so we will use 17 000 cm/s, which gives 0.017 cm/us.

$$Distance(cm) = TimerValue * 0.8704 \tag{4.5}$$

4.2 Tilt Angle

In this subsection, we will explain the process of measuring head's tilt angle. For this, we used an accelerometer and a microcontroller with an ADC. We decided to use ADXL335 (Figure 4.8). It has excellent temperature stability and low power consumption, what makes this device suitable for this project. In the state of the art, we presented the technical characteristics of this device.

Figure 4.8: ADXL335 module used [4]

It has an analogical interface so, to use it, the use of an ADC is required. Our microcontroller includes this feature what allows us to have a very simple hardware implementation. We only have to connect the accelerometer output to PIC ADC ports and place capacitors in each axis to select the bandwidth.

Table 4.1 shows the relation between the capacitor value and the sensor bandwidth.

Table 4.1: Table from ADXL335 datasheet representing the values of capacitors for specific bandwith

Bandwidth (Hz)	Capacitor (μF)
1	4.7
10	0.47
50	0.10
100	0.05
200	0.027
500	0.01

As we can see from Table 4.1, there are several options for bandwidth. In this dissertation, we decide to use 50Hz and so 0.10uF capacitors were added to each axis. If we intended to use the accelerometer to measure, for example, vibrations, we would choose a higher bandwidth because many readings would be required. For tilt angle measurement, 50Hz is suitable.

The firmware flowchart is presented in Figure 4.10. It starts with the ADC configuration. Then check the calibration of the device. If it is calibrated, start reading each axis and if not, we need to calibrate it.

Calibration is done by subtracting an offset to each axis. This device is ratiometric, which means that sensitivity varies proportional to supply voltage. From the datasheet [21], we can see that maximum value of sensitivity is 330mV/g. We know that for a flat position, we must have half of supply voltage and this value corresponds to the intended offset. Due to some noise, this calibration is done by empirical test. We have a reference value and change it until we read the correct values for flat position. After this is done, it is time to start reading each axis and then convert to acceleration values. This type of measurements are affected by the noise and to reduce this effect, we implemented software filters by taking several values and using an average.

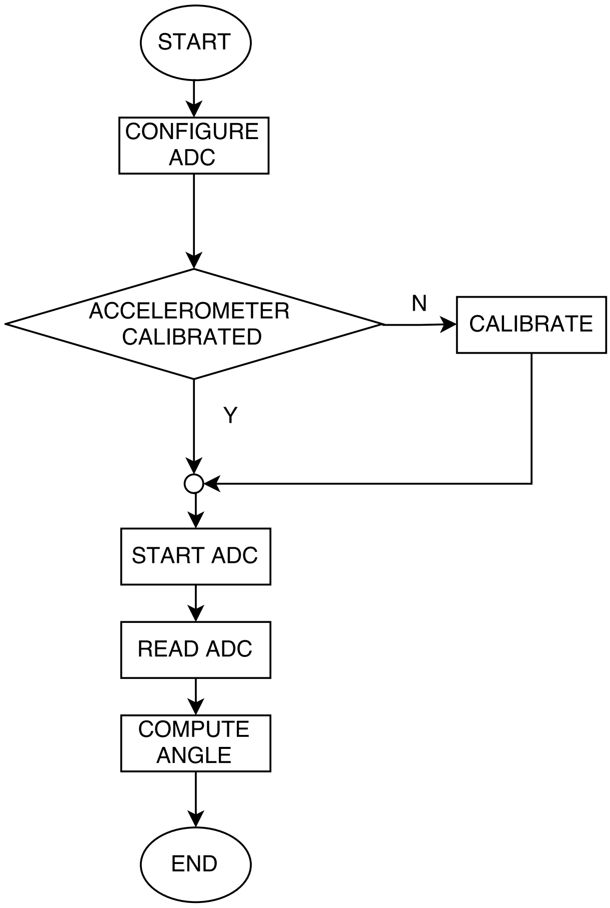


Figure 4.9: Firmware fluxogram to obtain head's tilt angle

The last step is to use the values read in both axis and convert them to degrees. To do this conversion, we have to understand the correspondent direction of each axis and use trigonometry rules. Figure 4.10 shows the defined referential for this accelerometer.

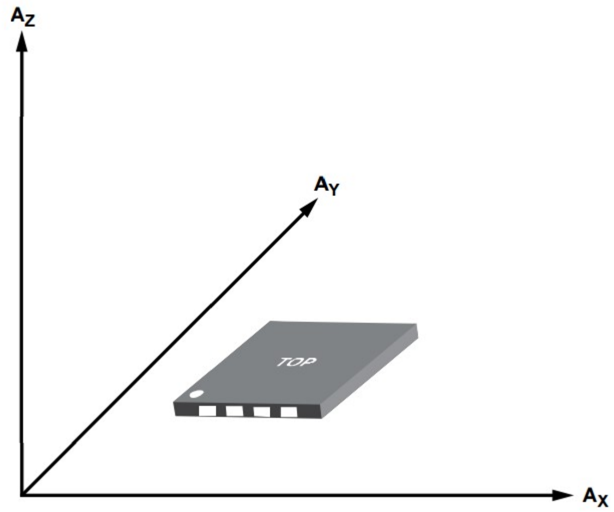


Figure 4.10: ADXL335 axis referencial

Since we are only working with x and z axis, to obtain the angle between them, a tangent is required. In this case, it is the ratio between acceleration in x and z, so before converting to degrees we have to transform the obtained values into acceleration. This is done using the sensitivity value mentioned before, 330mV/g where g represents the gravity acceleration. In ADC value, it corresponds to 102.3. First we have to subtract the offset value to calibrate and then we divide by 102.3 and we get the desired unit. It is supposed to have 0g and 1g at x and z axis respectively, when the accelerometer is in a horizontal position. The final degree value, as said before, is the ratio between accelerations in both axis:

$$angle = (arctan(\frac{Ax}{Az}) * 180)/\pi \tag{4.6}$$

This is the formula used to convert acceleration into degrees and it returns the inclination of the device, which will correspond to the head's tilt when applied directly on the animal.

Since we are using a 3 axis accelerometer, instead of using x and z axis, we could also use y and z. The formula is similar, replacing x for y, but the device must be placed in different position.

4.3 Communication with the radio interface

Communication between devices is one of the most important features of a virtual fence. So far we have designed the hardware and firmware to control the posture of each animal, which is one of the innovative characteristics of this project. Control animal's position in a predefined area without use of physical barriers and expensive GPS systems is what we want. To do that, we have to develop a localization system using radio frequency devices. This allows us to have a low power system and small devices, what provides us a more suitable system to place on animals.

For the communication between the base station and the animal it was used a Nordic NRF24L01+ RF module (Figure 4.11).

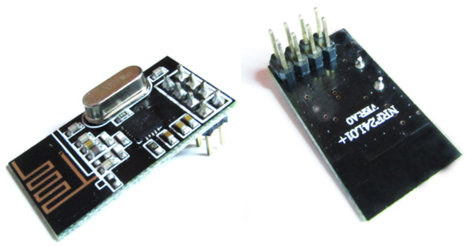


Figure 4.11: Nordic NRF24L01+ [5]

This module integrates a 2.4GHz RF transceiver with 250kbps, 1Mbps and 2Mbps on-air data-rate options. It has ultra low power consumption, which is one of the requirements of this project and it is compatible with other RF modules from the same manufacturer. NRF24l01+ is used in several applications such as PC peripherals, toys and consumer electronics [22].

In Figure 3.2 we can see that the communication between the RF transceiver and the microcontroller is done by SPI. Looking at the NRF24l01+ datasheet, we can find a table containing all SPI commands that can be used to interface both devices (Table 4.2).

Table 4.2: SPI commands from NRF24L01+ datasheet

Command name	Operation
R_REGISTER	Read command and status registers
W_REGISTER	Write command and status registers.
R_RX_PAYLOAD	Read RX-payload: 1 to 32 bytes.
W_TX_PAYLOAD	Write TX-payload: 1 to 32 bytes
FLUSH_TX	Flush TX FIFO
FLUSH_RX	Flush RX FIFO

Before put in practice a specific communication protocol, we must interface each RF module with the microcontroller. As said before, this interface is done by SPI. In the NRF24L01+ datasheet [23], we can find a table of registers that can be changed to configure the module with required features. In Table 4.3, we can see the available registers and the values we defined for them.

Table 4.3: NRF configuration values

Register	Value
CONFIG	0x0A
EN_AA	0x00
EN_RX_ADDR	0x01
SETUP_AW	0x01
SETUP_RETR	0x00
RF_CH	0x02
RF_SETUP	0x26
RX_PW_PO	0x06
RX_ADDR_PO	0xE7E7E7
TX_ADDR	0xE7E7E7

The **CONFIG** registers define if the radio is in transmitter or receiver mode, power up or power down. The first 4 bits define if we intend to work with interrupts. This module includes an active low IRQ pin that gets active when transmission or reception is complete.

The next register is **EN_AA**, related to a function called Enhanced Shockburst that this radio incorporate. It handles all transmissions, times and auto acknowledge. For this project, we want to control the transmission aspects by ourselves and, for that reason, this register is set to 0x00.

This radio includes up to 6 data pipes (communication channels) and **EN_RX_ADDR** is used to select one or more. In this case we only choose data pipe 0 for all slaves to transmit. Since we intend to use a TDMA communication protocol, we do not need to have a data pipe for each slave, which reduces the complexity of the system. After is necessary to define the width of address of data pipe 0 and that is done on register **SETUP_AW**. The RX and TX address has to be the same and we choose default with 3 bytes, 0XE7E7E7. For RX the register is **RX_ADDR_PO** and for TX is **TX_ADDR**.

RF_CH, **RF_SETUP** are used to configure radio properties like the frequency channel (there are 126 available), air data rate and output power from TX. For this project, without tests on outdoor, we set an air data rate of 250 Kbps and 0 dBm power.

The **SETUP_RETR** activates or deactivates automatic retransmissions. For this project, this is not necessary and for that reason we attributed 0x00 to this register. The last is **RX_PW_PO**. This one defines the width of the payload. It can be changed dependent on the type of message we want to send and it can go up to 32 bytes.

With all configurations done, it is time to transmit and receive data. The process is similar for both actions. This module has an active low interrupt pin that is activated when a transmission is completed or when it detects the arrival of a packet.

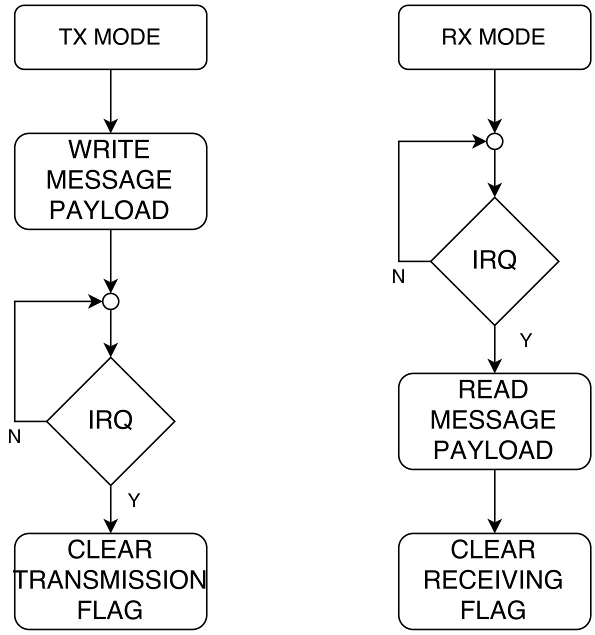


Figure 4.12: Diagram of the transmitting and receiving processes

In Figure 4.12, the process of transmitting and receiving is presented. Depending on what we intend to do, we must ensure the device is in the respective mode. For the transmission, we have to load the payload and when this is done, the IRQ pin goes low. The last step is to clear the transmission flag. For reception, the steps are similar, except we have to read the payload and clear the receiving flag.

4.3.1 TDMA communication protocol

With transmission and reception both implemented, it is time to start with the communication protocol. The master has to receive data from more than one slave and so, we must guarantee they do not transmit at the same time. As said in chapter 3, we decided to use a TDMA.

Time synchronization is extremely important in this communication protocol and so we need to know the transmission times in order to define the duration of each time slot. The size of the messages will also define the time of each slot.

Now we can transmit, receive and have defined how we will implement the communication protocol. So, to finish we have to start designing the master's and slaves's firmware.

Below, we will explain how we developed both firmwares with the help of software diagrams.

Figure 4.13 depicts a flowchart that illustrates the operation of the master. It starts by initializing the hardware and waiting for the beginning of a communication cycle, after what it broadcasts a synchronization message. When this transmission is concluded, the state changes to receiving and the module passes to rx mode. Now the master waits for the response of slaves. This is done by defining a time window and when that value is exceeded, the master returns to waiting state and the process restarts.

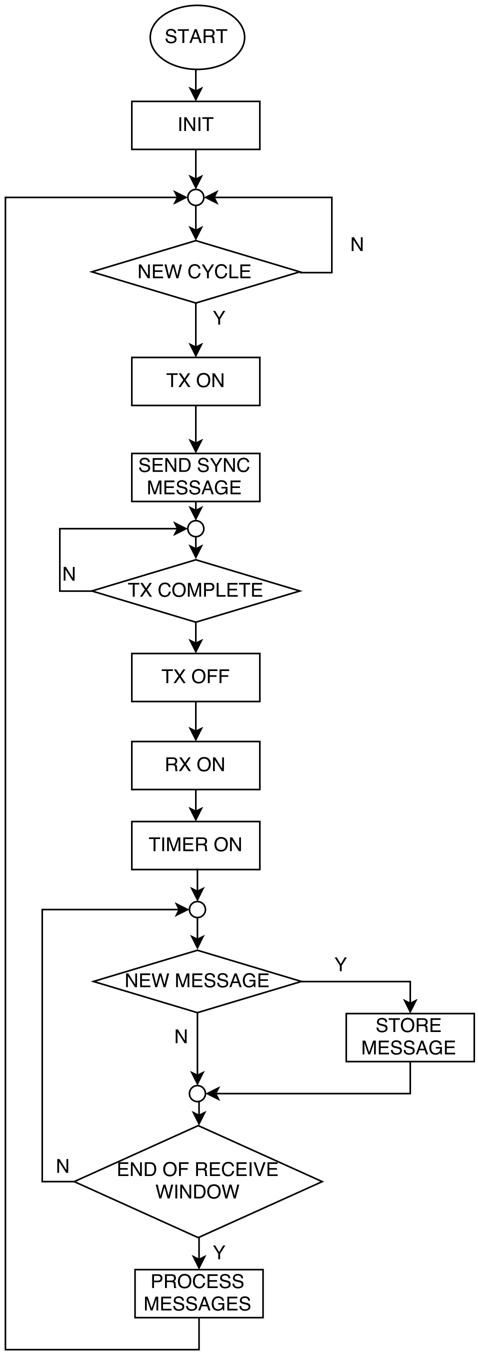


Figure 4.13: Master state flowchart

With the master implemented, we have to develop the slave firmware, illustrated in Figure 4.14. Initially, it starts by waiting for the master's synchronization message. At this time, all slaves are at the same step. After, each one has an attributed time slot and only in that time they start its transmission.

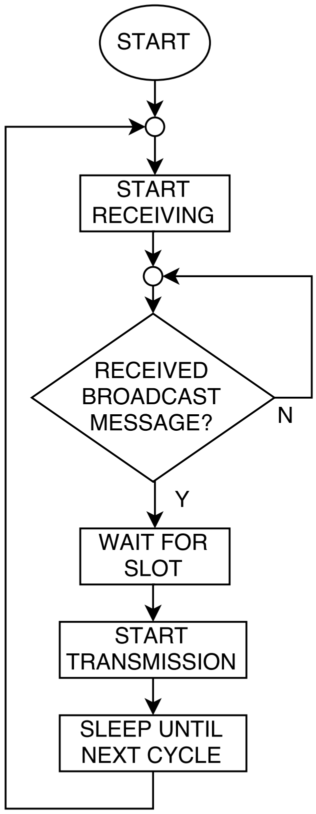


Figure 4.14: Flowchart of the slave firmware

Only after determining the transmission times, it is possible to define slot's time. With some tests, we know that the transmission of a message with a payload of 6 bytes takes, approximately, 600us. But we must take into account that this time corresponds only to loading the message to the payload and transmit it, we must add also the time that RF module take to pass from receiving mode to transmitting mode and the time that the microcontroller needs to communicate with the RF module. These values were not computed precisely, being used a slot time of 800us, which is a safe upper bound.

Since each slave has its own ID, the slots are defined depending on this value. To implement this protocol, we need to use two timers, one for slots and other for the entire communication cycle. Our microcontroller has 4 timers but only two are 16bits, which is necessary because a 8bit timer does not give us the required resolution. Timer 0 counts up to 3s, and we used this value as cycle time, this means, from 3 to 3 seconds, the master transmits a synchronization message. Timer 1 is used to attribute slots to slaves. It is activated after master's transmission and it is stopped when the receive window expires.

In Figure 4.15 the time diagram of the implemented TDMA with time information is presented.

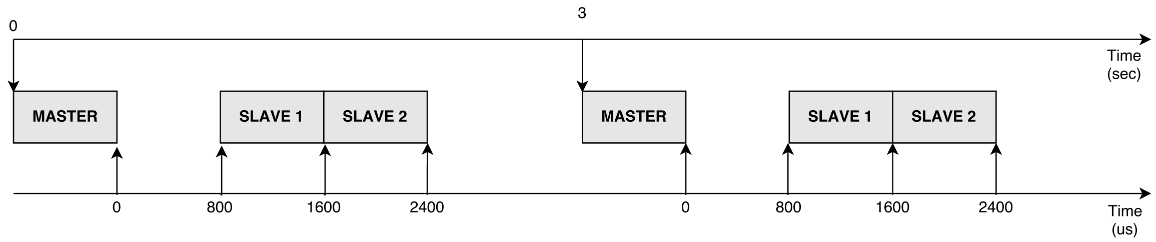


Figure 4.15: Time diagram of implemented TDMA with slots and cycle times information (2 slaves)

As said before, we defined a time slot of 800us. With this information, we attribute each slot depending on slave ID. Slave 1 starts at 800us and slave 2 at 1600us. We know that each transmission takes about 575us and so, we add more 225us to ensure that communications don't collide. This last value is an approximation and was not determined due to lack of time but this approach allows us to ensure that, if some delay occur, the slot time is enough for the complete transmission.

We performed several tests in order to test the effectiveness of this communication protocol and in next chapter we will present them and do an analysis of the obtained results.

Chapter 5

Results

This chapter presents the results of several tests that have been carried out to verify the correct operation of the different system blocks. It starts by distance to ground measurement, where we show how measurements were made and make some critical analysis to the respective results. The same procedure is then carried out for the accelerometer.

This chapter finishes with the results concerning the communication protocol, were parameters such as correctness of the message contents and the timeliness of the transmission is verified.

5.1 Distance to ground measurement

In this section we will present the results of distance to ground measurements using a transducer similar to the ones used in car parking systems. We know that these transducers can be used in outdoor spaces but this test was done in laboratory, where conditions are different from the ones where this device will be used. As explained before, the hardware developed for this part of the project was designed to measure in a 40 to 100 cm range. For values below or above this interval, no measurements can be done because it is not possible to detect echo and so, we can not measure time. In Table 5.1 and Figure 5.1, we present the measurements carried out. For each defined distance, we took 20 values and process them. With these, we calculated the error associated with each distance and also pointed out the minimum and maximum values obtained.

Table 5.1: Table containing minimum, maximum and error for each measured distance. All units in cm

Distance (cm)	**Min (cm)**	**Max (cm)**	**Standard Deviation**	**Mean (cm)**	**Error (cm)**
40	39	41	0.6070	40.5	0.5
50	50	52	0.6863	51.05	1.05
60	59	62	0.7592	60.55	0.55
70	68	72	0.8870	70.55	0.55
80	80	81	0.4104	80.8	0.8
90	90	91	0.4702	90.7	0.7
100	99	102	0.7327	100.3	0.3

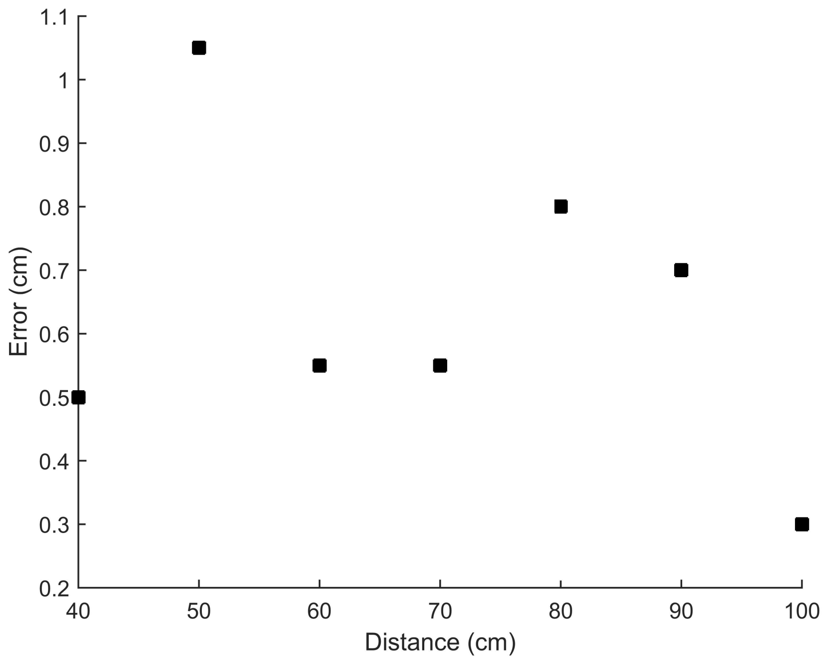


Figure 5.1: Graphic with the correspondent error values for each distance

From the graphic and table above, we can see that the maximum error obtained was for 50 cm. This ultrasonic sensor has a minimum sensing distance of approximately 40cm. Below this value, the echo is not detectable. From [24], we can see that, for a sheep with 20kg, the distance of the top of the head to the ground is 58cm, which leads us to conclude that probably, 40 cm of minimum range is not enough. This is a limitation of this system because it does not prevent the appearance, for example, of a rock on animal's path.

For the other distances, the error is lower than 1 cm. This value is acceptable because it allows us to detect, with a good accuracy, if the animal is trying to reach higher vegetation, which is one of the most important features of this part of the project.

It is important to refer that all hardware was implemented in breadboard. For certain, in a PCB, the error associated with each measurement will decrease because in a printed circuit board we have no wires and so, there is no risk of having crosstalk (phenomenon by which a signal transmitted on one circuit or channel of a transmission system creates an undesired effect in another circuit or channel [25]), what can happen in a breadboard. With this difference, the results are less affected by noise, what makes the echo detection more accurate.

In resume, the implemented system to measure the distance of the head to the ground using ultrasonic transducer cannot detect the presence of obstacles on the ground or if the animal is standing up or lying down because the minimum range obtained, 40cm, is not enough. In other hand, it can detect if the animal is trying to reach higher vegetation. In this situation, the measured distance will increase and we can get a maximum of 100cm, which is enough.

5.2 Tilt angle

As explained before, we intend to know the head's movements and it is possible by using more than one source of information. After knowing the distance to the ground, we have to determine how head is moving, this means tilt angle. Before we explained how this is done with an accelerometer and below the obtained results are presented in Table 5.2 and Figure 5.2.

Table 5.2: Table containing minimum, maximum, standard deviation, mean and error for each measured angle.

Angle (degrees)	**Min (degrees)**	**Max (degrees)**	**Standard Deviation**	**Mean (degrees)**	**Error (degrees)**
0	0	0	0	0	0
10	10	14	1.3169	11.5	1.55
20	18	24	1.6575	21.7	1.7
30	27	31	1.3139	30.4	0.4
45	39	46	3.2525	43.5	1.5
60	59	65	1.7313	59.95	0.05
70	70	75	1.9358	72.2	2.2
80	72	89	3.2298	80.7	0.7
90	87	92	1.8946	88.7	1.3

The table above contains minimum, maximum, standard deviation, mean and respective error for each measurement. In real situation, we can have two situations that we have to take into account. If the animal has its four legs on the ground, the maximum angle the head is tilted is 45 degrees. For this situation, we obtained a maximum error of 1.7 degrees, which is acceptable. The other is when the animal has two legs on the ground and the other two in a higher place, probably a tree or a rock. For this, the tilt angle can go up to 80 degrees and here we obtained a maximum error of 2.2 degrees, also acceptable.

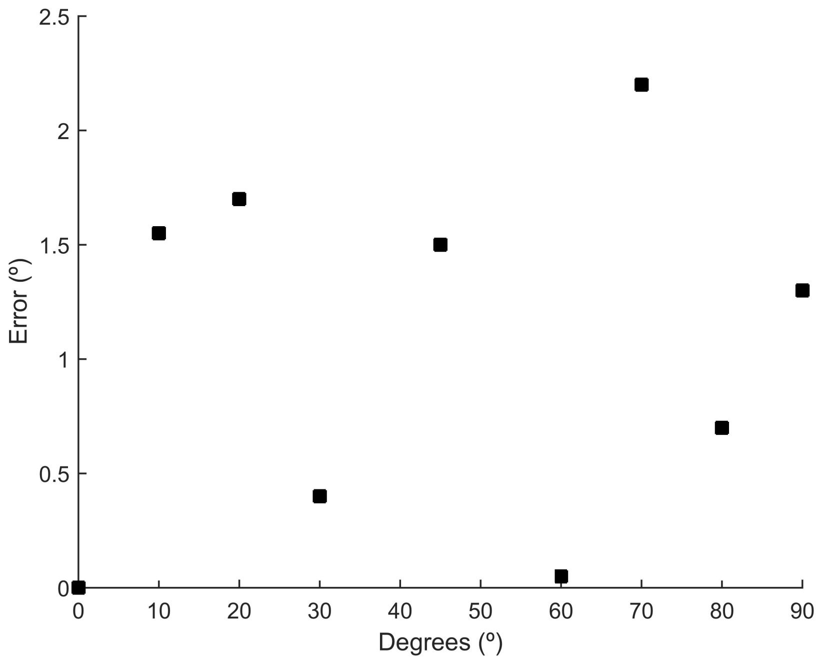


Figure 5.2: Graphic with the correspondent error values for each angle

The graphic above shows us how error varies when we move from 0 to 90 degrees. To improve the results, we used some filters. In terms of hardware, we added capacitors at each axis output to define the bandwidth to 50Hz and in software, we read ADC several times and use an average value of each measurement. To get to the most suitable value of readings, we tested several times with different values and we saw that for more than 20 times, the results were not better.

As same as for distance measurement with ultrasonic sensor, these results were obtained in laboratory. Obviously, this can change in outdoor spaces where temperature, humidity and other environmental changes can increase the error.

5.3 Communication Protocol

The last objective of this dissertation was to develop a communication protocol between base station and the animals. As explained before, all animals (slaves) have to communicate with a fixed pole (master). Since we will have a high number of slaves, we must ensure that we implement a system capable of avoiding collisions during transmissions and this was done with a TDMA. This protocol was explained before and in this section, we present the results obtained. Each animal has a collar with two sensors: an ultrasonic and an accelerometer. The first one measures the distance of head to ground and the second tilt angle. The master has to receive information from all slaves and for this, we defined that the message sent by slaves must include slave ID, distance and angle.

To test the effectiveness of this communication protocol, two slaves and one master communicated for two different periods of time and we measured if the pre-defined slot times were being respected.

The first thing done was to test if the transmission between each slave and master is working properly. To do this, we defined a sequence of values, distance and angle, for each slave and tested if this sequence has being correctly transmitted by each slave. We created two arrays of 10 values, presented in Table 5.3, with an increment of 10, what allow us to analyze in a more simple way the obtained results.

Table 5.3: Table containing values of distance and angle

Distance	10	15	20	25	30	35	40	45	50	55
Angle	20	25	30	35	40	45	50	55	60	65

Slave 1 transmits odd indexes from both arrays and slave 2 even indexes. In Tables 5.4 and 5.5 below we present the desired message sequence:

Table 5.4: Message sequence for slave 1

Message	ID	Distance	Angle
1	1	15	25
2	1	25	35
3	1	35	45
4	1	45	55
5	1	55	65

Table 5.5: Message sequence for slave 2

Message	ID	Distance	Angle
1	2	10	20
2	2	20	30
3	2	30	40
4	2	40	50
5	2	50	60

To test if each slave is transmitting the correct message sequence, the master sent the received values to a PC via serial port. The slaves are also connected to the PC, also via serial port.The results are presented in Figures 5.3 and 5.4. Each slave transmits after receiving the synchronization message from master and we expect to see the slave sending different messages after each master transmission.

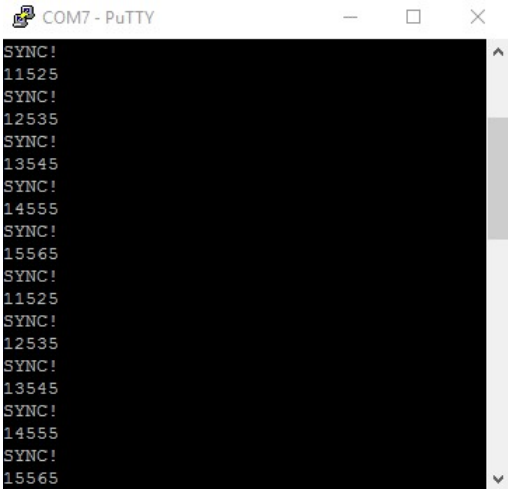


Figure 5.3: Slave 1 message transmission test

As we can see, slave 1 is working properly. It receives the synchronization message from master and, every time this happens, it sends different messages, following the intended message sequence defined before.

Figure 5.4: Slave 2 message transmission test

Slave 2 is also working as we defined and message sequence is being followed. With these results, we can conclude that slaves are receiving and transmitting as we intended to and for this reason, we can start analyzing the data received by the master.

As described before, the master is supposed to broadcast a message to all slaves and then, initiates a receiving window where all transmit a pre-defined message sequence. We have to test if master is receiving messages correctly and in defined times.

To do this, we tested the system for an half hour and then one hour, approximately, and then analyzed the number of messages that did not get to master. This first experiment was done in laboratory with master and slaves close together.

The way we defined message sequence allows us to determine in a easy way the number of lost messages. The next message is an increment of 10 and so, we only have to subtract consecutive messages and see if the result is 10.

For the second test, master and slaves were in different spots in the laboratory, separated by 5 meters. We expect to have a higher number of transmission errors because we will have greater influence from other devices that are being used in the room.

In the Table 5.6 below, we show the number of transmission errors in both test situations.

Table 5.6: Table containg the number of received and not received messages from both tests

Test	Messages	Received	Not Received	Success(%)
1	500	488	12	97.6
1	1000	982	18	98.2
2	500	479	21	95.8
2	1000	968	32	96.8

For both tests, we analyzed the effectiveness of the system for 500 messages and 1000 messages. It is possible to note that the percentage of success does not vary significantly from test 1 to test 2 and from 500 or 1000 messages. The difference between test 1 and 2 is the distance between slaves and master, and the effectiveness of this system is lower than test 1 but that difference is not relevant.

As described before, we developed a TDMA protocol to ensure that we have no collisions between slave's transmission. We have already tested if the messages are being sent correctly and if they are received with success by master, but we have to test also if the slots and cycle times are being respected. To do this, we started by measuring the time of each transmission in slaves.

In Figures 5.5 and 5.6 below, we show readings from the serial port where it is possible to see, after each transmission, the time correspondent to the end of transmission.

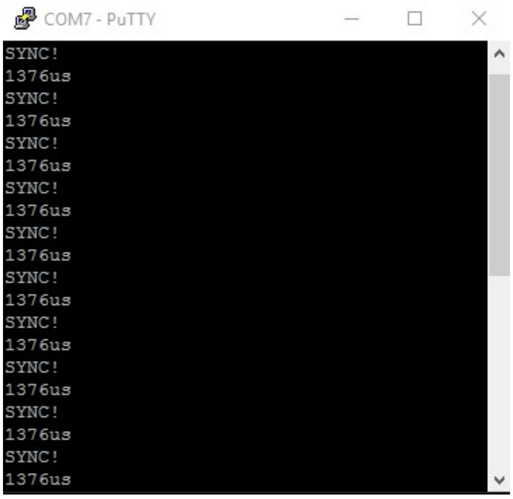


Figure 5.5: Slave 1 transmission time

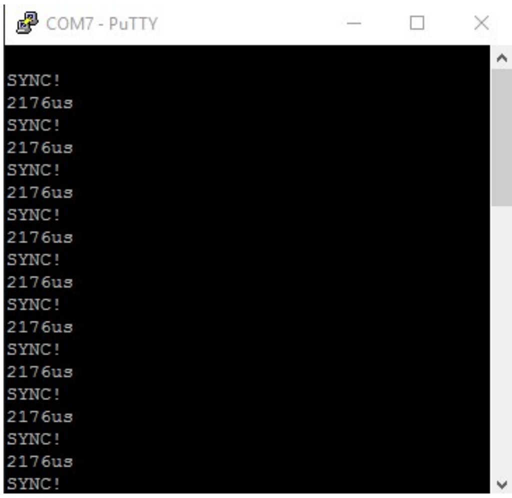


Figure 5.6: Slave 2 transmission time

As described in Figure 4.15, slave 1 starts the transmission when timer reaches 800us, which is the slot width. We know that each transmission takes about 575us and from Figure 5.5 we observe that slave 1 transmits in the correct time.

Slave 2 will start the transmission when timer value is 1600us and it is supposed to finish 575us after, which corresponds to the value presented in Figure 5.6.

With the confirmation of slaves transmission times, we have to test if master is receiving them in the correct slot. We tested the system with 100 messages and read the time that master receives the last data, which corresponds to the end of receiving window. Plotted in Figure 5.7, we present the time variation of each message.

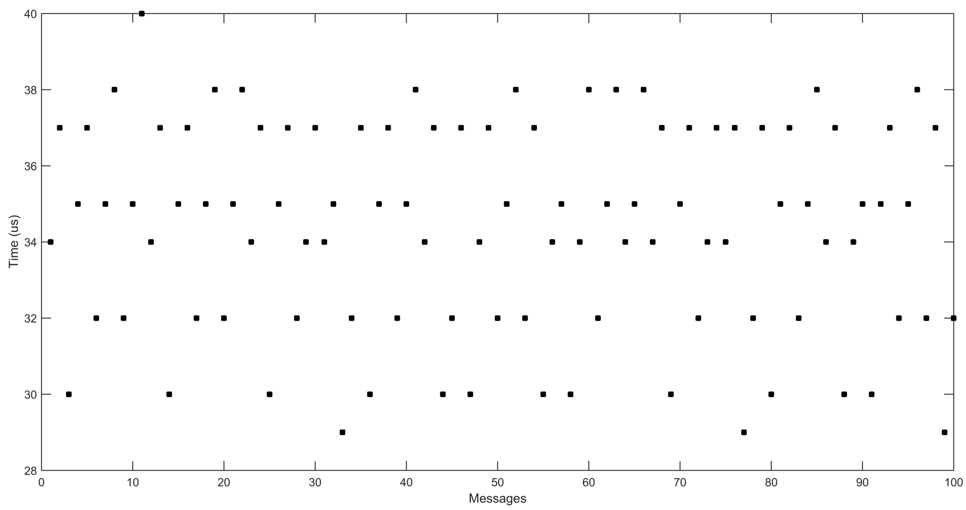


Figure 5.7: TDMA times variance for each message

It is possible to see that in 100 messages, the maximum error is 40us and the minimum is 29us. Obviously, this range of test does not reflect what will happen in the future when this system will be placed in different environment conditions and working for a larger amount of time, but for laboratory test, it shows the consistency of the implemented system.

From the presented results, we were able to establish communication between one master and two slaves using a TDMA protocol. We tested the correctiveness of the content of each transmitted message and the effectiveness of the pre defined times of the TDMA. We obtained good results from both, which leads us to conclude that the implemented communication protocol is ready to be applied in real situations.

Chapter 6

Conclusions and Future Work

The development of an animal behavior control mechanism was the main objective of this dissertation. We intended to develop a system that brings up new features to this concept like posture control and better localization system. For posture control, we needed a system that could measure the distance from animal's head to the ground in order to detect the movement of the animal, avoiding the animal from reaching unwanted vegetations. Obviously, with only this information, we were not able to gather the necessary data to decide if the animal is trying to reach them or not. So, we added an accelerometer to measure the angle the head is tilted and the fusion of both systems gives us a more confidant information about animals position.

Virtual fencing intends to confine animals in a pre defined area and for this reason we had to implement a localization system. Tough many solutions available use GPS, we decided to use radio frequency because we wanted a more efficient system in terms of accuracy and power consumption. The localization requires a communication protocol between a fixed pole (master) and the animals (slaves). Due to its efficiency and energy saving, we decided to implement a TDMA protocol, where each slave has a defined time slot to transmit, avoiding this way collisions. The transmitted messages from slaves will include information about animal like the head to ground distance, tilt angle and RSSI value. This last value is used to process the distance that the slave is from the master.

For the distance from the head to the ground measurement we decided to use an ultrasonic transducer similar to the ones used in car parking systems. These type of sensors are the most appropriate for this project because they work on the environmental conditions where we intend to apply this system, they are waterproof and humidity resistant. For this part and since we only have the transducer, we developed all hardware and firmware to interface the sensor. Posteriorly, we carried out some test the effectiveness and we can conclude that it can prevent all probable situations. For example, due to his minimum sensing range, which is 40cm, it can not detect the presence of objects between animals'head and the ground. On the other hand, it can be applied for avoiding the animal to reach higher vegetation because its maximum range is 100cm, which is enough for what we intend.

The angle the head is tilted was the second part of this project. As said before, we decided to add an accelerometer to measure the tilt angle to get a more reliable data about the posture of animal. It works as a complement of the information from ultrasonic transducer. Since our accelerometer included all hardware, we only designed the firmware and did some tests. We tested the system in a 0 to 90 degrees range and we obtained acceptable results, with a maximum error of 2.2 degrees.

The last part was the localization system. It was used a TDMA scheme, to save energy, and the results obtained show that the system works correctly, both in the time and value domains. Due to the lack of access to the RSSI value in the used RF module, we were no able to get the localization of each animal. But, the firmware developed can be applied to other module that can provide RSSI, and for that reason, all work done was not wasted. We have defined the messages to contain distance to ground, tilt angle and slave ID for master know which slave is transmitting.

As future work, fusing data from ultrasonic sensor and accelerometer could improve the reliability of this system. In this project, we have already defined messages to be able to transmit distance and angle at the same time. It is also necessary to use another RF module, as the one used in this work does not allow access to the RSSI value, thus, not allowing implementing the special confinement of herds.

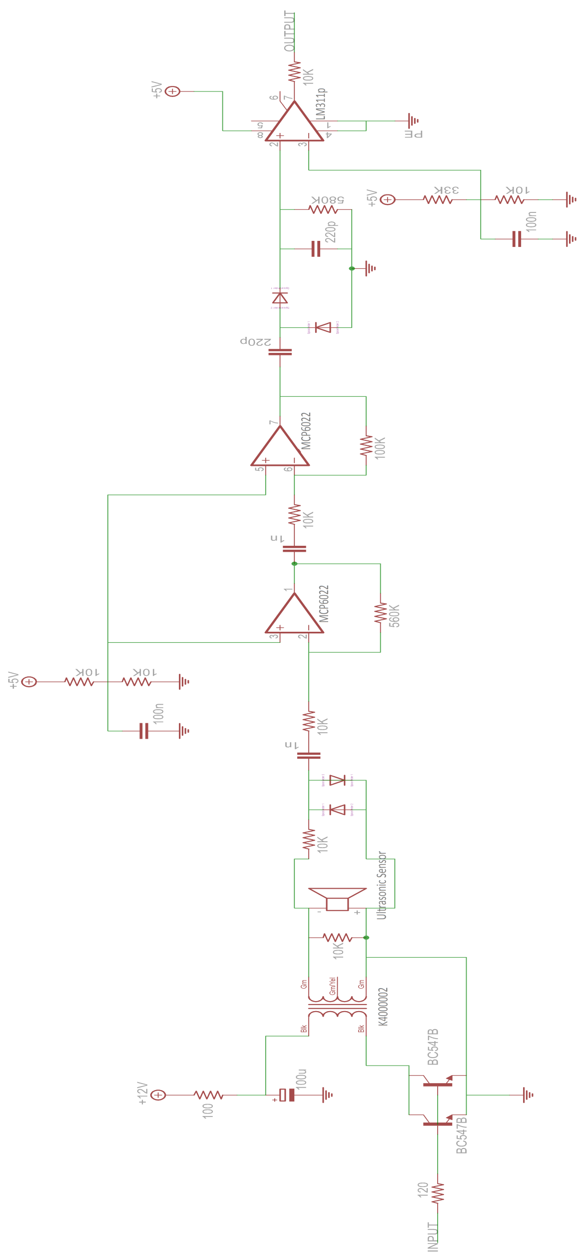
The development of the electric stimulator and warning buzzer circuitry has to be done in the future. It consists in designing a step up circuit, controlled by a microcontroller signal and that it would be easy to implement.

It is also necessary to perform tests outdoor to have get more reliable data, test the effects of different terrains and weathers when measuring the distance from the head to the ground and also the tilt angle. The conversion to PCB and the creation of the final collar is the next step, improving this way the efficiency of the entire system, specially reducing the minimum sensing range of the ultrasonic transducer.

The interface of the master with the herd manager is another point to implement in the future. There are several RF modules that incorporate bluetooth interface and with this functionality, it is possible to define that interface using this technology.

Appendices

Appendix A


OUTPUT
10K
LM311p
+5V
33K
10K
100n
580K
220p
220p
MCP6022
100K
10K
1n
MCP6022
560K
+5V
10K
10K
100n
10K
1n
10K
Ultrasonic Sensor
10K
K4000002
+12V
100
100u
BC547B
BC547B
120
INPUT

References

[1] RFM42B-868D RF Transmitter. http://static1.tme.eu/products_pics/f/1/a/f1a425f699458e09372b5da49c06fd33/40497.jpg. Accessed: 2016-10-29.

[2] FM-RRFQ1-433 RF Receiver. http://uk.farnell.com/productimages/standard/en_GB/1200971-40.jpg. Accessed: 2016-10-29.

[3] Ultrasonic Sensor. http://www.ekulit.de/wp-content/uploads/datasheets/700670%20K-14WP10.pdf. Accessed: 2016-10-29.

[4] ADXL335 module used. http://www.robotshop.com/media/catalog/product/cache/7/image/800x800/9df78eab33525d08d6e5fb8d27136e95/3/-/3-axis-accelerometer-module.jpg. Accessed: 2016-09-18.

[5] NRF24L01+. http://static.efetividade.net/img/n24l01figure01-30851.jpg. Accessed: 2016-09-15.

[6] Projecto self-prevention. www.self-prevention.com. Accessed: 2016-10-14.

[7] Instituto da vinha e do vinho. http://www.ivv.min-agricultura.pt/np4/125. Accessed: 2016-10-29.

[8] M. O. Monod, P. Faure, L. Moiroux, and P. Rameau. A virtual fence for animals management in rangelands. In *MELECON 2008 - The 14th IEEE Mediterranean Electrotechnical Conference*, pages 337–342, May 2008.

[9] Z. Butler, P. Corke, R. Peterson , and D. Rus. Dynamic virtual fences for controlling cows. pages 1–10, January 2004.

[10] Tim Wark, Peter Corke, Pavan Sikka, Lasse Klingbeil, Ying Guo, Chris Crossman, Phil Valencia, Dave Swain, and Greg Bishop-Hurley. Transforming agriculture through pervasive wireless sensor networks. *IEEE Pervasive Computing*, 6(2):50–57, April 2007.

[11] K. Heurtefeux and F. Valois. Is RSSI a Good Choice for Localization in Wireless Sensor Network? pages 732–739, March 2012.

[12] Jiuqiang Xu, Wei Liu, Fenggao Lang, Yuanyuan Zhang, and Chenglong Wang. Distance Measurement Model Based on RSSI in WSN. *Wireless Sensor Network*, 2:606–611, 2010.

[13] S. Lanzisera, D. Zats, and K. S. J. Pister. Radio frequency time-of-flight distance measurement for low-cost wireless sensor localization. *IEEE Sensors Journal*, 11(3):837–845, March 2011.

[14] V. Thotla, M. T. A. Ghasr, M. J. Zawodniok, S. Jagannathan, and S. Agarwal. Detection of super-regenerative receivers using hurst parameter. *IEEE Transactions on Instrumentation and Measurement*, 62(11):3006–3014, Nov 2013.

[15] Pedro Fonseca. Sistemas de Instrumentação Electrónica. Edição 2011/2012.

[16] Emily L. C. Shepard, Rory P. Wilson, Flavio Quintana, Agustina Gmez Laich, Nikolai Liebsch, Diego A. Albareda, Lewis G. Halsey, Adrian Gleiss, David T. Morgan, Andrew E. Myers, Chris Newman, and David W. Macdonald. Identification of animal movement patterns using tri-axial accelerometry. pages 1–14, March 2008.

[17] Ran Nathan, Orr Spiegel, Scott Fortmann-Roe, Roi Harel, Martin Wikelski, and Wayne M. Getz. Using tri-axial acceleration data to identify behavioral modes of free-ranging animals: general concepts and tools illustrated for griffon vultures. pages 986–996, October 2011.

[18] A beginners guide to accelerometers. `http://www.dimensionengineering.com/info/accelerometers`. Accessed: 2016-10-13.

[19] What is the difference between piezoelectric and piezoresistive components? `http://electronicdesign.com/components/what-s-difference-between-piezoelectric-and-piezoresistive-components`. Accessed: 2016-10-25.

[20] Accelerometer. `http://www.sensorwiki.org/doku.php/sensors/accelerometer`. Accessed: 2016-10-14.

[21] ADXL335 Datasheet. `https://www.sparkfun.com/datasheets/Components/SMD/adxl335.pdf`. Accessed: 2016-11-02.

[22] NRF24L01+ Module. `https://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01P`. Accessed: 2016-10-12.

[23] NRF24L01+ Datasheet. `https://www.sparkfun.com/datasheets/Components/SMD/nRF24L01Pluss_Preliminary_Product_Specification_v1_0.pdf`. Accessed: 2016-10-29.

[24] Herdwick Sheep Sculptures. `http://www.gardenartandsculpture.co.uk/portfolio-items/sheep-sculpture/`. Accessed: 2016-10-27.

[25] Crosstalk. `https://en.wikipedia.org/wiki/Crosstalk`. Accessed: 2016-10-29.